

PL/M-86

程序设计语言

(下册)

自动控制系

南京航空学院

一九八六年元月

第八章 8087 常用基本函数库

8087 支持库为 PL/M-86 运用浮点数运算提供了极大的方便，使它增添许多 FORTRAN 语言所具有的功能，才能使 NDP 成为一台精确而使用方便的浮点计算机。这是因为：

(1) NDP 是一台只能进行二进制运算的机器。尽管 NDP 能够理解多种数据类型，但它不能直接对浮点十进制数进行运算，必须调用软件，先把十进制数转换成二进制形式，运算完再将二进制的结果转换成十进制的输出。

(2) 8087 的指令集仅局限于已为它实现了的一些基本函数。例如虽有正切函数，却无正弦函数，而且正切的输入仅限于 0 到 $\pi/4$ 之间。一个具有完整定义域的完备基本函数集有待于软件来实现。

(3) 非屏蔽的异常情况处理使得使用软件复杂化。有了支持库就方便了。

因为支持库由 PL/M-86 (或 ASM-86) 程序调用，就需使用 PL/M 86 编译程序。由于支持库的所有过程均为 FAR 过程，所以应在 MEDIUM 或 LARGE 计算模型下编译 PL/M-86 模块。还须用 LINK 86 和 LOC86，将用户程序生成的目标块转换成可装入内存或编程于内存 (或 ROM 中) 的可执行程序。

在调用 8087 时必须对 NDP 系统进行初始化，必须装入中断 20 至 31 的地址，在接口库文件 INIT87 和 INITFP 中都提供了 8087 初始化过程。8087。LIB 库文件对 8087 部件进行初始化，并设置控制字。E8087。LIB 库文件在 8087 的初始化之前，装入中断

20 至中断 31。87NULL.LIB 库文件不给 8087 提供任何外部过程。只能由不包含 NPX 指令的软件使用。（NPX —— 数字处理器扩展部件）。

INIT87 和 INITFP 之间的区别在于 8087 控制字的缺省设置。INIT87 屏蔽所有的异常，INITFP 屏蔽除“1”以外的所有异常。尽管以后可能要对控制字进行修改，用户还是应该首先调用 INIT87 或 INITFP。

本章描述 CEL87.LIB 中的各组数均为常用的浮点运算函数。8087 芯片对每个 CEL87 函数中最费时间的部分给出了直接的支持但函数的形式及输入的值域也受到限制。CEL87 提供了必要的软件以使 8087 能够支持一个完善的基本函数包，对所有适当的输入都能给出正确的结果。

8.1 CEL87 函数库

按功能把 CEL87 函数分组如下列出。

8.1.1 舍入和截断函数

mqrIEX 将实数舍入成与其接近的整数，若有两相近的整数，则将其舍入成偶数，取回结果为实型。

mqrIE2 除结果为 16 位整数外，其它同 mqrIEX。

mqrIE4 除结果为 32 位的整数外，其它同 mqrIEX。

（PL/M-86 不能用）。

mqrIAX 将实数舍入成离零较远的整数。取回的结果值为实数。

mqrIA2 除结果为 16 位的整数外，其它同 mqrIAX。

mgerIA4 除结果为 3 2 的整数外，其它同 mgerIAX。(PL / M-86 不能用)。

mgerICX 截去实数的小数部分。结果为实数。

mgerIC2 除结果为 1 6 位的整数外，其它同 mgerICX。

mgerIC4 除结果为 3 2 位的整数外，其它同 mgerICX。(PL / M-86 不能用)。

8. 1. 2 对数与指数函数

mger LGD 计算数常用对数值(以 10 为底)。

mger LGE 计算自然对数(以 e 为底)。

mger EXP 计算以 e 为底的指数。

mger YZX 计算以任意数为底的指数。

mger YI2 求实数输入的 A X 次整数幂。

mger YI4 除输入整数为 DXAX 外，其它同 mger YI2。

(PL / M 不能用)。

mger YIS 除输入整数放于 8086 栈顶外，其它同 mger YI2。

8. 1. 3 三角函数与双曲函数。

mger SIN、mgerCOS、mgerTAN 分别计算正弦、余弦和正切。

mgerASN、mgerACS、mgerATN 分别计算上述函数的反函数。

mgerSNH、mgerCSH、mgerTNH 分别计算上述函数的双曲函数。

mqerAT2 这是反正切函数的一种特殊形式，它在二维空间内进行直角坐标至极坐标的象限转换。

8. 1. 4 其它函数

mqerDIM FORTRAN的正差函数。

mqerMAX 计算两个实数输入中的最大值。

mqerMIN 计算两个实数输入中的最小值。

mqerSGN 将一个输入的符号与另一个输入的幅值拼凑在一起。

mqer求模MOD 保留被除数的符号。(PL/M-86中已有此运算符)。

mqerRMD 求模，取离零近的结果值。

所有CEL87的过程名都使用了四个字母“mqer”。加这个前缀是为了避免同用户程序中使用的其它名相冲突。用小写是为突出后面的函数过程名。

8. 2 PL/M-86程序中对CEL 87过程的说明。

在用户编写PL/M-86程序时当要调用CEL86基本函数库中的各函数过程，都必须在用户程序开头的说明语句中加以说明，这样才能调用。

```
mqerACS, PROCEDURE(X) REAL EXTERNAL;
```

```
    DECLARE X REAL;
```

```
    END mqerACS; /*反余弦函数*/
```

```
mqerASN PROCEDURE (X) REAL EXTERNAL;
```

```
    DECLARE X REAL;
```

```

        END mger ASN;    /*反正弦函数*/
mger AT2 PROCEDURE(Y,X) REAL EXTERNAL;
        DECLARE(Y,X) REAL;
        END mgerAT2;    /*特殊正切函数*/
mger ATN: PROCEDURE(X) REAL EXTERNAL;
        DECLARE X REAL;
        END mger ATN;  /*反正切函数*/
mgerCOS: PROCEDURE(THETA) REAL EXTERNAL;
        DECLARE THETA REAL;
        END mger COS;  /*余弦函数*/
mgerCSH: PROCEDURE (THETA) REAL EXTERNAL
        DECLARE THETA REAL;
        END mger CSH; /*双曲余弦函数*/
mger DIM PROCEDURE (Y,X) REAL EXTERNAL;
        DECLARE(Y,X) REAL;
        END mger DIM; /*FORTRAN的正差函数*/
mgerEXP: PROCEDURE(X) REAL EXTERNAL;
        DECLARE X REAL;
        END mger EXP; /*计算以e为底的指数*/
mger IA2, PROCEDURE(X) INTEGER EXTERNAL;
        DECLARE X REAL;
        END mgerIA2; /*舍入取整函数*/

```

mgerIAX: PROCEDURE(X) REAL EXTERNAL;

DECLARE X REAL;

END mgerIAX; /*舍入函数*/

mger IC2; PROCEDURE (X) INTEGER EXTERNAL;

DECLARE X REAL;

END mgerIC2 ; /*截断取整函数*/

mger ICX: PROCEDURE (X) REAL EXTERNAL;

DECLARE X REAL;

END mgerICX; /*截断函数*/

mger IE2; PROCEDURE (X) INTEGER EXTERNAL;

DECLARE X REAL;

END mger IE2; /*舍入取整函数*/

mger IEX PROCEDURE (X) REAL EXTERNAL;

DECLARE X REAL;

END mger IEX ; /*舍入函数*/

mger LGD: PROCEDURE (X) REAL EXTERNAL;

DECLARE X REAL;

END mger LGD ; /*10为底常用对数*/

mger LGE: PROCEDURE (X) REAL EXTERNAL;

DECLARE X REAL;

END mger LGE; /*e为底对数*/

```
mger MAX: PROCEDURE (Y, X) REAL EXTERNAL;
    DECLARE (Y, X) REAL;
    END mger MAX; /*最大值函数*/
```

```
mger MIN: PROCEDURE (Y, X) REAL EXTERNAL;
    DECLARE (Y, X) REAL;
    END mger MIN; /*最小值函数*/
```

```
mger MOD: PROCEDURE (Y, X) REAL EXTERNAL;
    DECLARE (Y, X) REAL;
    END mger MOD; /*求余数*/
```

```
mger RMD: PROCEDURE (Y, X) REAL EXTERNAL;
    DECLARE (Y, X) REAL;
    END mger Y2X; /*计算任意底的指数*/
```

```
mger YIS: PROCEDURE (Y, I) REAL EXTERNAL;
    DECLARE Y REAL I INTEGER;
    END mger YIS;
```

```
END mger IEX; /*舍入函数*/
```

```
mger LGD: PROCEDURE (X) REAL EXTERNAL;
    DECLARE X REAL;
    END mger LGD; /*10为底常用对数*/
```

```
mger LGE: PROCEDURE (X) REAL EXTERNAL;
    DECLARE X REAL;
    END mger LGE; /*e为底对数*/
```

```
mger MAX: PROCEDURE (Y, X) REAL EXTERNAL;
    DECLARE (Y, X) REAL;
    END mger MAX; /*最大值函数*/
```

```
mger MIN: PROCEDURE (Y, X) REAL EXTERNAL;
    DECLARE (Y, X) REAL;
```

```
      END mger MIN; /*最小值函数*/
mger MOD; PROCEDURE (Y, X) REAL EXTERNAL;
      DECLARE (Y, X) REAL;
      END mger MOD; /*求余数*/
mger RMD; PROCEDURE (Y, X) REAL EXTERNAL;
      DECLARE (Y, X) REAL;
      END mger RMD; /*取余数*/
mger SGN; PROCEDURE (Y, X) REAL EXTERNAL;
      DECLARE (Y, XX) REAL;
      END mger SGN; /*X的符号与Y的值合并*/
mger SIN; PROCEDURE (THETA) REAL EXTERNAL;
      DECLARE THETA REAL;
      END mger SIN; /*正弦函数*/
mger SNH; PROCEDURE (THETA) REAL EXTERNAL;
      DECLARE THETA REAL;
      END mger SNH; /*双曲正弦函数*/
mger TAN; PROCEDURE (THETA) REAL EXTERNAL;
      DECLARE THETA REAL;
      END mger TAN; /*正切函数*/
mger TNH; PROCEDURE (THETA) REAL EXTERNAL;
      DECLARE THETA REAL;
      END mger TNH; /*双曲正切函数*/
mger Y2X; PROCEDURE (Y, X) REAL EXTERNAL;
      DECLARE (Y, X) REAL;
```

8.3 在 PL/M-86 程序中怎样调用 CEL87 函数

CEL87 函数对 8087 栈中的数进行操作，不过在用户用 PL/M-86 编程序时是看不到这一点的，没有什么 PL/M-86 语句可以直接往 8087 栈里存数或取数。这样的操作在调用 REAL 数运算时隐含地产生的。

给出说明语句后，调用 CEL87 函数时只要将其放在赋值语句中就可以了。所有经过说明的 CEL87 函数都会使编译程序在遇到 CEL87 调用时将输入的参数入栈，并在 CEL 过程结束时将结果放到正确的目

的地址。

PL/M-86不支持LONG-REAL和TEMP-REAL格式。PL/M所使用的常数和变量均为32位的SHORT-REAL数，置于8086的内存。当这些数被压入8087栈时，它们都被转换成TEMP-REAL数。在PL/M将保存在8087栈中的OPL87函数结果返回时，函数值又被从TEMP-REAL转换成SHORT-REAL，转换是由编译程序所生成的FSTP指令完成的。（FSTP指令为从8087栈中取出数存到内存（或另一个栈之）并弹栈操作）。

8.4 8087库基本函数的功能说明和举例

1. mger ACS(X) $y = \arccos(X)$ 反余弦函数。

功能： $y = \text{mger ACS}(X)$ 。当输入值在-1到1之间，则输出 y 为弧度值在0至 π ， x 、 y 均为实型数。

2. mger ASN(X) (arcsine)

输入参数为8087栈项数

功能：取回一个以弧度表示的数，该数的正弦等于输入值。正确输入，在-1到+1之间。

结果的值域为 $-\pi/2$ 到 $\pi/2$ 。

3. mger ATN(X) $y = \arctan(x)$

输入参数 (x) 为8087栈项数

功能：mger ATN取回一个用弧度表示的数，其正切值等于输入值。取回值限于 $-\pi/2$ 至 $\pi/2$ ，输入为无穷值为有效的。

4. mger AT2(y, x) (直角坐标转换为极坐标函数)

$$X = \arctangent(y/x)$$

输入参数：(x)为8087栈顶数；(y)为8087栈顶第二个数。

功能：mger AT2执行把直角坐标到极坐标的 θ 角的转换。可把(x)、(y)解释为直坐标平面上一个点的坐标，则mger AT2取的是用弧度表示的从正x轴转到该点的角度如图8-1。

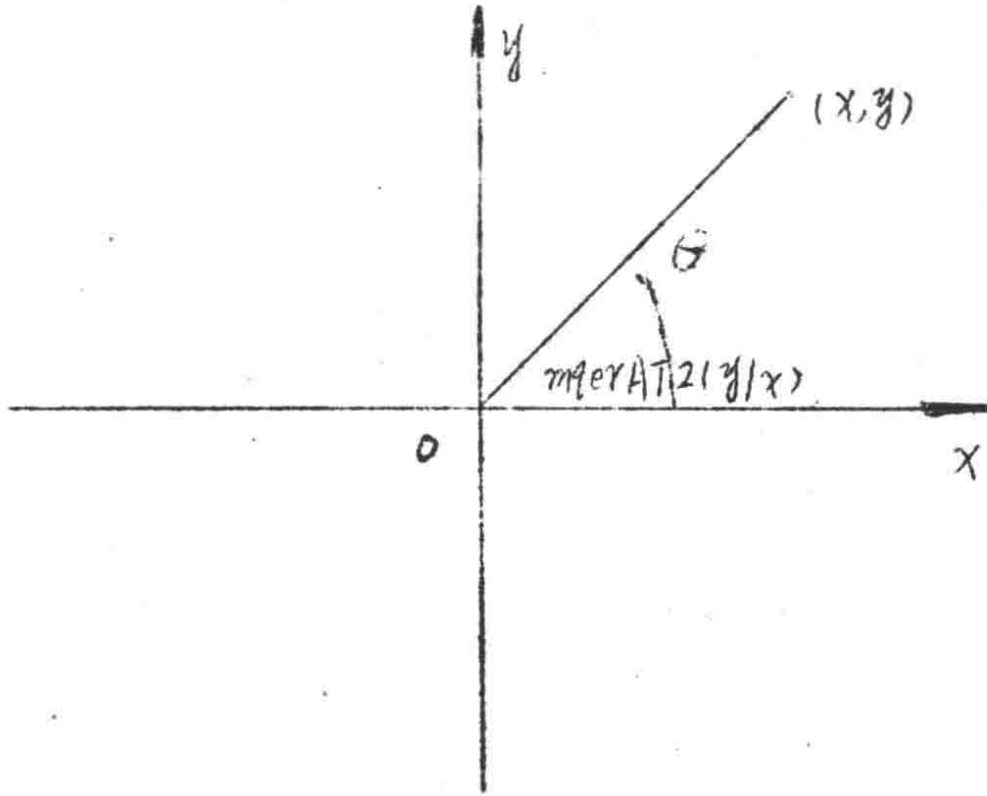


图 8-1 直角坐标到极坐标的转换

角的范围从 $-\pi$ (负 x 轴下附近的点) 到 π (负 x 轴上或上方附近的点)。

角的计算公式如下

若 $(x) > 0$ 则求 $(\text{arcTangentT}(y/x))$ 。

若 $(x) = 0$ 则求 $\text{mqer SGN}(\pi/2, Y)$ (见此函数解释)

若 $(x) < 0$ 则求 $(\text{arcTangentT}(y/x) + \text{mqerSGN}(\pi, y))$

在 (x, y) 中允许一个为无穷值。当 $x = \text{INFINITY}$ 结果值为 0 并带 y 的符号当 $x = -\text{INFINITY}$ 得结果值为 π 并带 y 的符号; 当 $y = \text{INFINITY}$ 则结果值为 $\pi/2$, $y = -\text{INFINITY}$ 得 $-\pi/2$ 。

输出: 8087 栈弹出一个元, 结果值取代了两个输入值。

5. $\text{mqer COS}(x) \quad x = \text{COSine}(x)$

输入参数: (x) 为 8087 栈顶数

功能: mqer COS 取 (x) 的余弦, 其中 (x) 是用弧度表

示的角度。对输入的零、伪零及非规格化数结果均为1；对于 2^{-60} 的未规格化数也得1。

例1 求 $\cos(x)$ ， $\sin^{-1}(y)$ ， $\cos^{-1}(x)$ ， $\tan^{-1}(y)$ 等函值。DO，

```

DECLARE TOKEN LITERALLY
$ INCLUDE( RMX86 / INC / CBWIR. EXT)
DECLARE CR LITERALLY 'ODH', LF LITERALLY
'OAH', PI LITERALLY '3.1415926538979';
DECLARE (x, y1, y2, y3, y4) REAL,
DECLARE (X1, X2, X3, THETA) REAL DATA(5.0,
10.0, 30.0, 60.0), (CON EXCEP)WORD,
mqer ACS: PROCEDURE(x) REAL EXTERNAL,
        DECLARE X REAL,
        END mqerACS,
mqer ASN: PROCEDURE(X) REAL EXTERNAL,
        DECLARE X REAL,
        END mqerASN,
mqer AT2: PROCEDURE(y, x) REAL EXTERNAL,
        DECLARE (y, x) REAL,
        END mqer AT2,
mqer ATN: PROCEDURE (x) REAL EXTERNAL:
        DECLARE X REAL,
        END mqer ATN:

```

```
mquer COS: PROCEDURE (THETA) REAL EXTERNAL;  
    DECLARE THETA REAL;  
    END mquer COS;
```

```
CALL INIT$REAL$MATH$UNIT;
```

```
X = P I / 1 8 0;
```

```
Y 1 = X * X 3;
```

```
Y 4 = mquerCOS ( Y 1 );
```

```
Y 2 = X * THETA ;
```

```
Y 3 = mquerCOS(Y 2);
```

```
/*输出实型数 Y 4, Y 3. 即  $\cos(30^\circ)$ ,  $\cos(60^\circ)$  的值
```

```
*/
```

```
X = 1 8 0 / P I;
```

```
Y 1 = X 1 / X 2;
```

```
Y 2 = mquer ASN ( Y 1 );
```

```
Y 1 = mquer ACS ( Y 1 );
```

```
Y 3 = X 1 / X 1;
```

```
Y 3 = mquer ATN(Y 3);
```

```
Y 4 = mquer AT2(x 1, X 2);
```

```
/*输出 Y 1, Y 2, Y 3, Y 4, 即输出  $\cos^{-1}(0.5)$ ,
```

```
 $\sin^{-1}(0.5)$   $\tan^{-1}(1)$  的弧度值和输出  $y = 5$ 。  $x = 10$  的
```

```
极坐标弧度值*/
```

```
Y 1 = Y 1 * X;
```

```
Y 2 = Y 2 * X;
```

```
Y 3 = Y 3 * X;
```

$y_4 = y_4 * X;$

/*输出 y_1, y_2, y_3, y_4 , 即输出角度值*/

END;

附此程序在机上运行结果。

\$LARGE

\$OPTIMIZE(3)

CP:DO;

DECLARE TOKEN LITERALLY "WORD";

\$INCLUDE(RRMX86/INC/CBWIR.EXT)

DECLARE (CON EXCEP) WORD;

DECLARE CR LITERALLY 'ODH' ;PI LITERALLY

'3.1415926538979' LF LITERALLY 'QAH';

DECLARE (X,Y1,Y2,Y3,Y4) REAL,

(X1,X2,X3,THETA) REAL DATA(5,10.,30.,60.);

mgerASN:PROCEDURE(X) REAL EXTERNAL;

DECLARE X REAL;

END mgerASN;

mgerAT2: PROCEDURE(Y,X) REAL EXTERNAL;

DECLARE (Y,X) REAL;

END mgerAT2;

mgerATN: PROCEDURE(X) REAL EXTERNAL;

DECLARE X REAL;

END mgerATN;

mgerACS: PROCEDURE(X) REAL EXTERNAL;

```
DECLARE X REAL

END mgerACS;

mgerCOS: PROCEDURE(THETA) REAL EXTERNAL;

DECLARE THETA REAL;

END mgerCOS;

CALL INIT$REAL$MATH$UNIT;

CON=OPEN$LP;

X=PI/180;

Y4=X*X3; /* X, 转为弧度值 */

Y3=X*THETA; /* 0 转为弧度值表示 */

Y4=mgerCOS(Y4);

Y3=mgerCOS(Y3);

    CALL REAL$SEND(CON, Y4);

    CALL CRLF(CON);

CALL REAL$SEND(CON, Y3);

CALL CRLF(CON);

X=180/PI; /* 1 弧度的度数 */

Y1=X1/X2;

Y2=Y1;

Y3=X1*X2;

Y1=mgerACS(Y1);

Y2=mgerASN(Y2);

Y3=mgerATN(Y3);

Y4=mgerAT2(x1, x2);
```

CALL REAL\$SEND(CON, Y1);

CALL CRLF(CON);

CALL REAL\$SEND(CON, Y2);

CALL CRLF(CON);

CALL REAL\$SEND(CON, Y3);

CALL CRLF(CON);

CALL REAL\$SEND(CON, Y4);

CALL CRLF(CON);

Y1=Y1*X;

Y2=Y2*X;

Y3=Y3*X;

Y4=Y4*X;

CALL REAL\$SEND(CON, Y1);

CALL CRLF(CON);

CALL REAL\$SEND(CON, Y2);

CALL CRLF(CON);

CALL REAL\$SEND(CON, Y3);

CALL CRLF(CON);

CALL REAL\$SEND(CON, Y4);

CALL CRLF(CON);

CALL CLOSE(CON);

CALL RQ\$EXIT\$IOS\$JOB(EXCEP, 0, @ EXCEP);

END CP;

输出运行结果:

- 8. 66025E-01 /* COS 30° 的值 */
- 5. 00000E-01 /* COS 60° 的值 */
- 1. 04720E+00 /* 60° 的弧度值 */
- 5. 22599E-01 /* 30° 的弧度值 */
- 1. 55080E+00 /* Tan⁻¹ (50) 的弧度值 */
- 4. 63648E-01 /* Tan⁻¹ (0.5) 弧度的值 */
- 6. 00000E+01
- 3. 00000E+01
- 8. 88542E+01
- 2. 65651E+01

6. mgerCSH(X) 双曲余弦 X=hyperbolic Cosine(x)

输入参数: (x) 为 8087 栈顶数 (实型) 用弧度表示的角度

功能: mgerCSH 取 (x) 的双曲余弦, 所有输入的零, 伪零和非规格化的数均得结果值 1。同样对小于 2⁻⁶ 的未规格化数也得 1, 正负无穷值均得值 + INFINITY, 不发生错误。

7. mger DIM (y, x) x = max (Y - X, +0)

输入参数: (x) 为 8087 栈顶数, (Y) 为 8087 栈顶第二个数。

功能: mgerDIM 先比较 (Y) 与 (X) 若 (X) > (Y) 则结果为 +0, 若 (Y) > (X) 则取正值 (Y - X)

当 8087 芯片工作于仿射 (有符号无穷值) 方式且两个输入不相等时, 允许输入为无穷值, 结果仍从以上的比较与减法运算中获得。